

main.py

```
import my_module  
  
a = 1  
print(my_module.f(a))
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

main.py

```
import my_module  
  
a = 1  
print(my_module.f(a))
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main *

main.py

```
import my_module # *  
  
a = 1  
print(my_module.f(a))
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main *

main.py

```
import my_module # *  
  
a = 1  
print(my_module.f(a))
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main

G: my_module *

main.py

```
import my_module  
  
a = 1  
print(my_module.f(a))
```

my_module.py

```
import math # *  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main

G: my_module *

"math" → (module)

main.py

```
import my_module  
  
a = 1  
print(my_module.f(a))
```

my_module.py

```
import math  
  
a = math.sin(1) # *  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main

G: my_module *

```
"math" -> (module)  
"a"     -> 0.8414709848078965
```

Ramki wykonania

main.py

```
import my_module  
  
a = 1  
print(my_module.f(a))
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x): # *  
    return x + a
```

B: built-in

...

G: main

G: my_module *

```
"math" -> (module)  
"a"     -> 0.8414709848078965  
"f"     -> (function)
```

main.py

```
import my_module  
# *  
a = 1  
print(my_module.f(a))
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main *

"my_module" → (module)

G: my_module

"math" → (module)
"a" → 0.8414709848078965
"f" → (function)

main.py

```
import my_module  
  
a = 1 # *  
print(my_module.f(a))
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main *

```
"my_module" -> (module)  
"a"          -> 1
```

G: my_module

```
"math" -> (module)  
"a"    -> 0.8414709848078965  
"f"    -> (function)
```

Ramki wykonania

main.py

```
import my_module  
  
a = 1  
print(my_module.f(a)) # *
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main *

```
"my_module" -> (module)  
"a"          -> 1
```

G: my_module

```
"math" -> (module)  
"a"    -> 0.8414709848078965  
"f"    -> (function)
```

Ramki wykonania

main.py

```
import my_module  
  
a = 1  
print(my_module.f(a)) # *
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main

```
"my_module" -> (module)  
"a"          -> 1
```

G: my_module *

```
"math" -> (module)  
"a"    -> 0.8414709848078965  
"f"    -> (function)
```

L: my_module.f *

main.py

```
import my_module  
  
a = 1  
print(my_module.f(a))
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x): # *  
    return x + a
```

B: built-in

...

G: main

```
"my_module" -> (module)  
"a"          -> 1
```

G: my_module *

```
"math" -> (module)  
"a"    -> 0.8414709848078965  
"f"    -> (function)
```

L: my_module.f *

```
"x" -> 1
```

Ramki wykonania

main.py

```
import my_module

a = 1
print(my_module.f(a))
```

my_module.py

```
import math

a = math.sin(1)

def f(x):
    return x + a # *
```

B: built-in

...

G: main

```
"my_module" -> (module)
"a"          -> 1
```

G: my_module *

```
"math" -> (module)
"a"     -> 0.8414709848078965
"f"     -> (function)
```

L: my_module.f *

```
"x" -> 1
return value: 1 + 0.8414709848078965
```

Ramki wykonania

main.py

```
import my_module  
  
a = 1  
print(my_module.f(a)) # *
```

my_module.py

```
import math  
  
a = math.sin(1)  
  
def f(x):  
    return x + a
```

B: built-in

...

G: main

```
"my_module" -> (module)  
"a"         -> 1
```

G: my_module *

```
"math" -> (module)  
"a"    -> 0.8414709848078965  
"f"    -> (function)
```

Output: 1.8414709848078965